Real-time Holoscopic 3D VIDEO INTERLACING

1st Yizhou Huang
College of Engineering
Department of Electronic and Computer Engineering
Brunel University, London, UK
Yizhou.Huang@brunel.ac.uk

2nd Mohammad Rafiq Swash
College of Engineering
Department of Electronic and Computer Engineering
Brunel University, London, UK
Rafiq.Swash@brunel.ac.uk

3rd Abdul Sadka
College of Engineering
Department of Electronic and Computer Engineering
Brunel University, London, UK
Abdul.sadka@brunel.ac.uk

***Abstract*— With the continuous development and improvement of display technology, the plane reality technology has been unable to meet user's demand and expectation. Because 2D image lacks the depth of field and unable to read Z-axis information as known as 3D depth. The three-dimensional display technology can not only solve this problem, but also overcome technical limitations, while providing a multi-view 3D viewing experience, but also as much as possible to restore colour, image clarity and resolution of 3D display like 2D display. On the other hand, three-dimensional display technology still exists some issues such as content's acquisition and image pre-processing processes are complex and lack to directly compatible with the traditional 2D content. This paper proposes a real-time Holoscopic image rendering method. This method brings a 3D model to Holoscopic based 3D content and interlaces all images as a sequence in real-time. Other than multi-view 3D image process, depending on the number of viewpoints, real-time Holoscopic 3D interlacing method services 3D image under a single step instead of a multi-camera process to acquire image sequence to the composite 3D image. In addition, this method is suitable for 3D image rendering with different viewpoints by changing the number of pre-rendered cameras, then synthesizing the image sequence to generate a new Holoscopic 3D image to save times for making 3D video.**

Keywords*— Holoscopic, Integral image, Real-time Rendering, lenticular, Lens array, Pixel mapping, Unidirectional, 3D display, Viewpoint

I. Introduction

Holoscopic 3D imaging also as known as integral imaging is a light field imaging system that mimics fly's eye technique to acquire spatial scene using a single aperture camera using a micro lens array (MLA). A holoscopic 3D image is composited of thousands of micro images which all contributes for true 3D reconstruction of the 3D scene. A holoscopic 3D display of the same MLA specification must be used to visualize the acquired holoscopic 3D content as a single aperture true 3D scene is reconstructed by all 3D cubes which are projected via a MLA.

Computer graphics rendering technology enables the Holosopic 3D visualization system to run in a virtual environment. 3D contents are obtained by calculation by constructing a virtual holoscopic camera array.

This article focuses on how holoscopic achieves fast rendering and real-time monitoring under computer virtual imaging technology. It proposes a real-time rendering method that converts multiple 3D format files into supported 3D format. In which 3D format file generates a resource library in a unique holoscopic 3D environment through a virtual camera array written in code. Where contents can be grouped in the repository and rendered in real time via the interlaced holoscopic video player.

II. Relateed work

Stereoscopic 3D display system [1] set two cameras in the same plane to simulate the distance between the human eye, capturing images resembling the left and right eyes, and synthesizing the left and right images by wearing the device to watch the 3D display. Autostereoscopic 3D Multiview display [2] based on stereoscopic 3D display [1], bringing more viewing angle changes and motion parallax[3][4], as shown in Figure 1, using the camera array, through it captures objects and scene, render the eight elemental images of the camera, and finally composite as a 3D image. Unlike the above, the Holoscopic 3D display system [4][5][6] simulates the fly's eyes, and the array of cameras is arranged as shown in Figure 2.

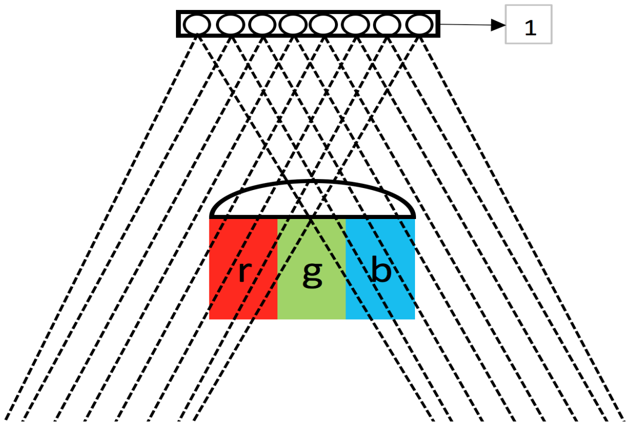

Figure 1 Autostereoscopic multi-view camera array (8 views) (1) Horizontal camera arrays.

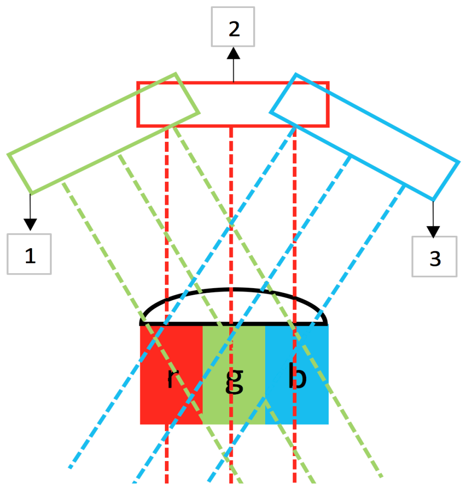

Figure 2 Illustration of Holoscopic 3D camera, different viewpoints, (1) camera_1, (2) camera_2, (3) camera_3.

In the rendering process, although most of the 3D software supports algorithms and code writing, multi-view image can be rendered by setting a camera array key-generated bottom, but this rendering method saves the image in the camera's default write File, which is unable to realize real-time inspection of all images, in other words, that is fail to do real-time rendering and checking. This paper presents a real-time rendering method, through the format conversion to transfer 3ds max and other 3D production software file into obj format file, and then import into Pov-ray for real-time rendering. The Pov-ray workflow is shown in Figure 3.

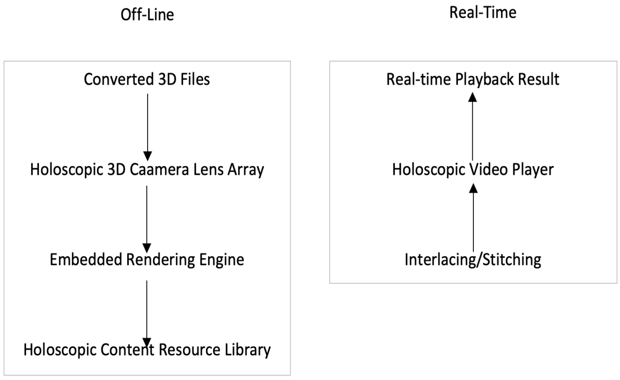

Figure 3 Workflow of pre-processing of Pov-ray.

III. PREPARATION OF HOLOSCOPIC CAMERA LENS ARRAY

Holoscopic 3D display provides a 3D resolution of 2D width resolution divide by the number of viewpoints multiply by height resolution, for instance, in this paper a 13 viewpoints Holoscopic 3D display is designed and developed to display 3D result with a 3830 (width) * 2160 (height) high resolution monitor. Thus, horizontal 3D resolution equal 3840/13=295, and height resolution unchanged. Algorithm is designed to set the number of cameras – as known as number of viewpoints, lens focal length, display width and height which is equal the dimensions of display and most importantly to control the lends width of Holoscopic camera.

Lens width of Holoscopic camera is used to match different pixel mapping techniques [7], in addition, when lenticular lens or parallax barrier [8][9] technology is applied on the display, where the lens width of Holosocpic camera is equal each micro lens or the pinhole of parallax barrier as figure 4 shown below. But this paper is focus on real time Holoscpic rendering and interlacing method and therefore, more details of the relationship between Holoscopic camera’s lens width and lenticular lens or parallax barrier technique will cover other subject.

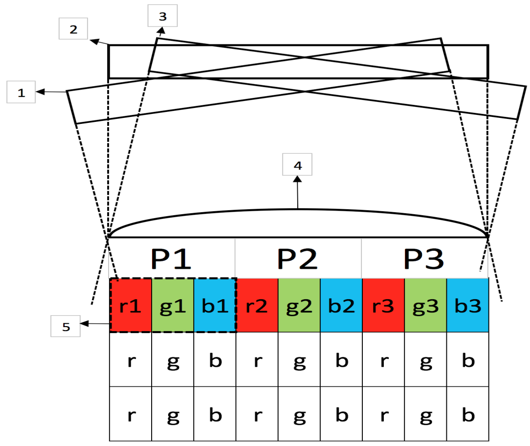

Figure 4 Lenticular lens pixel mapping and Holoscopic camera layout. (1-3) number of cameras. (4) micro lens width, (5) RGB colour pixel.

This real-time rendering method saving render time and meanwhile can be observed in real time to check the changes between objects, in a timely manner to make the appropriate adjustments to meet the needs of 3D display. Figure 5 illustrates the overall workflow of stitching Holoscopic 3D content and figure 6 provides pseudocode to refer the implementation principle.

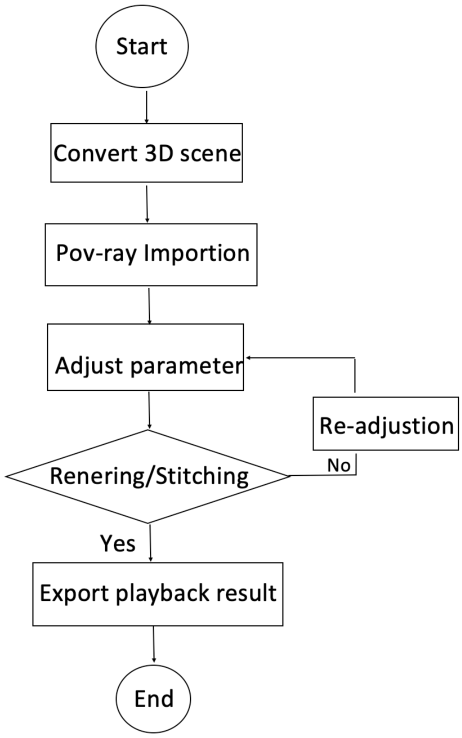

Figure 5 Overall progress of Holosocpic 3D content interlacing.

```
Example:
// All H3D contents have to be interlaced under 1000 milliseconds
   1. Assign computer hard drive to read 13 viewpoints H3D contents
      If contents have more or less than 13 images
         Print "failed"
      Else
         Print "Stitch Pixel Level"
   2. Use time reader to count rendering time
      If total render time under 1000 milliseconds
         Print "Save ResultingH3D Image"
      Else
         Print "failed"
   3. Create Rendering repot and go back to beginning.
```

Figure 6 Pseudocode of algorithm of counting interlaced content

We have designed and developed a prototype in the lab to test the Holoscopic 3D image and compare rendering time of both multi-view and Holoscopic 3D to verify this method has capability to save rendering time. The compared playback result and rendering time will be mentioned on the next section. The specifications of prototype as shown in table1.

Items	Specification
Display dimensions	625.92 (W)*352 (H) mm
Pixel pitch	0.163 (H) * 0.163 (V) mm
Pixel per lens	13
Lens pitch	2.12 mm
2D resolution	3840 (H) *2160 (V) pixels
Lens number	295 lens

Table 1 Specifications of Holoscopic 3D prototype, designed and developed by Brunel University.

IV. REAL-TIME HOLOSCOPIC 3D IMAGES RENDERING AND STITCHING

Real-time Holoscopic 3D rendering method and settings as it discussed above, brings 13 images as a sequence to be able to interlace 13 viewpoint images as a new 3D image. When we get real-time rendering of the image, and determine the image is correct, the 13 images will be interleaved, the principle of interleaving as shown in Figure 7. The first row and line of each image is the initial position, and then the start point of 13 images are interleaved together to form the first row and line of the new 3D image, and so on, the computer do progressive scanning and render the 13 images all intertwined together, as shown in Figure 8.

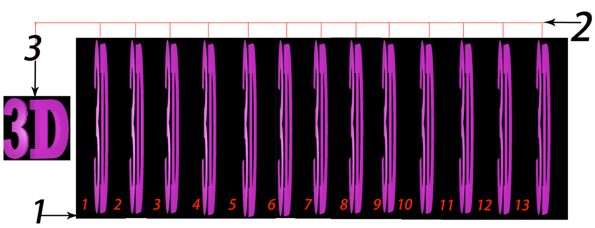

Figure 7 Interlaced 13 viewpoints images, (1) Each red line in vertical direction contains 1 pixel in horizontal direction, (2) Each image contains 295 pixels in horizontal direction, (3) Stitched 3D image which contains 295 *13= 3840 pixels in horizontal direction.

Figure 8 Rendered 13 viewpoints 3D image.

For instance, the horizontal resolution of each image is 295, where the horizontal direction contains 295 pixels, each pixel contains RGB sub-pixels, as the result, the new 3D image after interleaving contain 295 $*$ 13 horizontal pixels, that is 3840 Pixel (image pixel rounding, the actual pixel for each image is 295.3846, approximately after the integer is 295, so 295 $*$ 13 $\approx$ 3840, rounding the value does not affect the playback result of 3D display).

The real-time rendering method can check the motion parallax of each view image during the rendering process. In other words, when the adjacent pictures have excessive motion parallax, the rendering process can be stopped in time. We take the reference object, for example, too much 3D depth of field cause ghosting, virtual coke and other issues, so each 3D scene has a maximum depth of field, the front and after of reference object represents the maximum positive and negative parallax[3][4][10] of the 3D scene, When we use the real-time rendering method for 3D rendering, if the scene of the object moves beyond the reference object, that is more than the correct display of the maximum parallax, we can stop rendering in a timely manner, without waiting for the entire rendering process after the end of the inspection Or modify it. As shown in Figure 9, in the whole animation process, the object moves out of the reference object, thus, we can timely adjust object's position, get the correct rendering results.

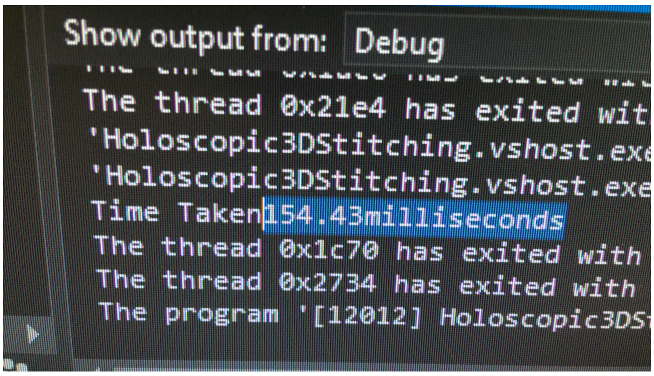

Figure 9 Rendering time per frame of Holocopic 3D video player, frame rate: 4fps, total rendering time 154.43 milliseconds.

In general, by using the real-time rendering method, we can effectively reduce the rendering time, and can frequently check the object's parallax, modify and amend duly. Table 2 shows the entire rendering of this test video.

From the table, it is easily to find, a total of 120 frames (frame rate is 4) use 1020 seconds when the 3D video is arrived. As figure 9 indicates that the average rendering time per frame is approximate 0.154 seconds, which is a very exciting result.

	Views	Views	Views	Views
Pixels per Lens	4	7	9	13
Frame Rate	4	4	4	4
Total Frame	120	120	120	120
Total Rendering time/second	18.48	20.64	22.44	24.24
Rendering time of per frame/second	0.154	0.172	0.187	0.202

Table 2 Total rendering time and specifications

V. CONCLUSION

In this paper, we proposed and developed a real time holoscopic 3D video interlacing method which enables to replay holoscopic 3D content on the light field display. The playback interlacing is 30 frame per second and the proposed method takes all holoscopic 3D viewpoint images and interlace them to reconstruct a single holoscopic 3D image which is replayed at 4 fps. In addition, the result indicates that with the number of viewpoints increases, the number of pixels per view in the horizontal direction decreases, which cause the rendering speed per unit time faster, but these two reference quantities are not linear correlation, this provides a good idea for subsequent research.

ACKNOWLEDGMENT

We gratefully acknowledge the support of NVIDIA Corporation with the donation of the Tesla K40 GPU used for this research.

REFERENCES

[1] H. J. Choi, ‘Current status of stereoscopic 3D LCD TV technologies’, *3D Res.*, vol. 2, no. 2, pp. 1–4, 2011.

[2] Y. Zhang, Q. Ji, and W. Zhang, ‘Multi-view autostereoscopic 3D display’, *OPEE 2010 - 2010 Int. Conf. Opt. Photonics Energy Eng.*, vol. 1, pp. 58–61, 2010.

[3] Y. Zhao, Y. Wei, ‘Parallax Polarizer Barrier Stereoscopic 3D Display Systems’, *IEEE/OSA J. Disp. Technol.*, vol. 3, no. 11, pp. 1165–1168, 2013.

[4] M. R. Swash, A. Aggoun, O. Abdulfatah, B. Li, J. C. Fernandez, and E. Tsekleves, ‘Holoscopic 3D image rendering for Autostereoscopic Multiview 3D Display’, *IEEE Int. Symp. Broadband Multimed. Syst. Broadcast. BMSB*, pp. 1–4, 2013.

[5] A. Aggoun, E. Tsekleves, M. R. Swash, D. Zarpalas, A. Dimou, P. Daras, P. Nunes, and L. D. Soares, ‘Immersive 3D holoscopic video system’, *IEEE Multimed.*, vol. 20, no. 1, pp. 28–37, 2013.

[6] A. Agooun, O. A. Fatah, J. C. Fernandez, C. Conti, P. Nunes, and L. D. Soares, ‘Acquisition, processing and coding of 3D holoscopic content for immersive video systems’, *3DTV-Conference*, pp. 5–8, 2013.

[7] T. J. Packard, R. Cited, U. S. P. Documents, J. Eloise, and J. J. Hohenshell, ‘Ulllted States Patent [19]’, vol. 39, 1999.

[8] R. De La Barre, R. Bartmann, M. Kuhlmey, B. Duckstein, S. Jurk, and S. Renault, ‘A new design and algorithm for lenticular lenses display’, *2016 Int. Conf. 3D Imaging, IC3D 2016 - Proc.*, 2016.

[9] M. R. Swash, A. Aggoun, O. Abdulfatah, J. C. Fernandez, E. Alazawi, and E. Tsekleves, ‘Distributed pixel mapping for refining dark area in parallax barriers based holoscopic 3D Display’, *3D Imaging (IC3D), 2013 Int. Conf.*, no. 1, pp. 1–4, 2013.

[10] P. Surman, I. Sexton, K. Hopf, W. K. Lee, E. Buckley, G. Jones, and R. Bates, ‘European research into head tracked autostereoscopic displays’, *2008 3DTV-Conference True Vis. - Capture, Transm. Disp. 3D Video, 3DTV-CON 2008 Proc.*, vol. 1, no.1, pp. 161–164, 2008.